VOD 시스템에서의 가변 비트율 MPEG 비디오 저장 및 검색 기법의 성능 평가

전용희[†] · 박정숙[††]

요 약

주문형 비디오 시스템에서, 비디오 데이터는 일반적으로 자기 디스크 배열에 저장된다. 데이터 검색을 위한 실시간 요구사항을 제공하기 위하여 연속적인 매체의 전달이 시간적으로 보장되도록 비디오 스트림들이 클라이언트들에게 연속적으로 전달되어야 한다.

프로세서와 네트워크의 증가된 성능에 비교하면, 자기 디스크 시스템의 성능은 단지 약간만 개선되었을 뿐이다. 저장 시스템의 성능을 개선하기 위하여 디스크 배열 시스템이 제안되어 사용되고 있다. 배열 시스템은 디스크들을 병렬로 배치하고 데이터를 동시에 검색함으로써 입출력 성능을 개선한다.

본 논문에서는, VOD 시스템에서의 비디오 데이터를 접근하기 위하여 고정 시간 길이(CTL : Constant Time Length)와 고정 데이터 길이(CDL: Constant Data Length) 두 가지 액세스 정책을 고려하였다. 디스크 스케줄링 정책도 그 두 개의 범주로 분류하고, 데이터 액세스 정책과 디스크 스케줄링 정책을 동시에 고려한 통합 환경에서, 디스크 배열의 동기화 정도에 따른 최대 허용 비디오 스트림수에 대하여 비교하였다.

비교된 스케줄링 정책 중에서 LOOK이 제일 우수함을 보여주고 있고, 동기화 정도별로는 동기화가 커질수록 이득이 있었다. CTL과 CDL의 성능 비교에서는 CTL이 최대 허용 스트림 수 측면에서 약간 우수한 성능을 보여줌을 확인할 수 있었다.

Performance Evaluation of VBR MPEG Video Storage and Retrieval Schemes in a VOD System

Yong-Hee Jeon[†] and Jung-Sook Park[††]

ABSTRACT

In a VOD(Video-On-Demand) system, video data are generally stored in magnetic disk array. In order to provide real-time requirements for data retrieval, video streams must be delivered continuously to the clients such that the delivery of continuous media can be guaranteed in a timely fashion.

Compared to the increased performance of processors and networks, the performance of magnetic disk systems have improved only modestly. In order to improve the performance of storage system, disk array system is proposed and used. The array system improves I/O performance by placing disks in parallel and retrieving data concurrently.

In this paper, two approaches are considered in order to access the video data in a VOD system, which are CTL(Constant Time Length) and CDL(Constant Data Length) access policies. Disk scheduling policies are also classified into the two categories and compared in terms of the maximum allowable video streams with different degrees of disk array synchronization, under the mixed environments in which both data access policy and disk scheduling policy are considered.

Among the compared scheduling policies, LOOK was shown to have the best performance. In terms of degree of disk synchronization, more gain was achieved with large degree of synchronization. In comparisons of performance of CTL and CDL, CTL was proved to have a little superior performance in terms of number of maximum allowable streams.

이 논문은 1997년 한국학술진흥재단의 공모과제 연구비에 의하여 연구되었음.

[†] 정회원, 대구가톨릭대학교 공과대학 컴퓨터정보통신공학부
[††] 준회원, 한국전자통신연구원 인터넷기술연구부

1. 서 론

사용자들의 요구가 단순한 정보 서비스에서 멀티미디어 서비스에 대한 요구로 변하는 추세와 더불어, VLSI 기술, 영상 처리 기술, 광대역 종합 정보 통신망(B-ISDN : Broadband Integrated Services Digital Networks) 등의 관련 산업 기술의 발달로 통신망을 이용하여 영상 서비스를 제공할 수 있는 새로운 멀티미디어 서비스 시대의 도래가 가능하게 되었다. 이 대표적인 서비스 시스템으로 주문형 비디오(VOD : Video On Demand) 시스템이 개발되고 있다[2,5-7,9,10, 12,13,15].

VOD 시스템은 사용자의 요구에 따라 요청된 서비스를 통신망을 통하여 제공하고, 사용자가 서비스를 제어할 수 있는 시스템이다. VOD 시스템은 비디오, 오디오, 그래픽 그리고 텍스트 등의 디지털 데이터를 개별적으로 혹은 결합된 멀티미디어 정보를 압축하여 비디오 서버에 저장하고, 사용자가 요청한 서비스를 통신망을 통하여 사용자에게 서비스를 제공하는 시스템으로 사용자는 프로그램 혹은 서비스를 선택, 제어할 수 있으며, 필요한 정보를 검색, 질의할 수 있는 양방향 대화형 서비스를 제공하는 시스템이다. 제공되는 서비스로는 Movie On Demand, News On Demand, 대화형 게임, 홈 쇼핑, 전자식 백과사전, 교육 등이 있다[6].

VOD 시스템의 구성요소로는 비디오 서버, 네트워크, 셋톱(Set-Top) 박스 등이 있다. 비디오 서버는 각종 정보를 저장 · 제공하는 시스템으로 연속적인 스트림들의 프로그램물을 많은 시청자에게 동시에 전달한다는 점에서 종래의 파일서버와는 차이가 있다. 비디오 서버의 내용물들은 통상적으로 하드디스크에 저장된다. 그러므로 디스크 관리와 파일 구조는 종래의 파일서버와는 다르게 되며, 대규모의 저장 용량이 필요하다[2-5,9,17,18,21,25-28]. CPU 기술의 발달로 프로세서의 성능과 디스크 속도 사이의 차이는 점차로 넓어지고 있는 실정이다. 그러므로, VOD 시스템에서 디스크 시스템이 대표적인 병목 현상을 초래할 수 있다.

VOD 시스템에서 VOD 서버는 비디오 저장소 및 비디오 펌프로 구성된다. 비디오 펌프의 처리속도에 비해 저장소에서 해당 데이터를 검색하는 속도가 훨씬 느리므로, 비디오 펌프의 이용률과 시스템 전체적인 입장에서 수용 가능한 최대 스트림수는 저장소의 검색성능에 의해 제한된다.

비디오 저장소의 성능을 연구하기 위해서는 크게 두 가지 관점을 고려할 수 있다. 첫째는, 비디오 저장소의 성능이 저장소의 물리적인 특성에 의해 상한값이 결정되므로 이러한 한계를 극복하기 위한 여러 대안들을 사용하는 것이다. 둘째는, 이러한 물리적인 특성에 의한 상한값들을 바탕으로 사용 가능한 여러 기법들의 효율성을 최대화시키자는 것으로, 스케줄링 기법, 액세스 정책, 블록배치기법과 같은 사항들이 있다.

일반적으로 VOD 시스템에서 저장매체로 사용되는 것에는 RAM 외에 자기테이프, 자기디스크, 광디스크와 같은 매체들을 고려할 수 있지만 비용 효과적인 측면에서 자기디스크가 가장 적합한 매체로 고려되고 있다. 특히 디스크 시스템의 성능 향상은 여러 개의 디스크를 병렬로 연결한 디스크 배열을 사용함으로써 가능하다. 그러므로 본 논문에서는 저장매체 중 자기디스크를 대상으로 성능분석을 수행한다.

VOD 서비스를 위한 요구사항 중에 실시간성 만족과 동시수용 스트림수의 최대화가 있다. VOD 서비스의 실시간성 만족을 위해서 VOD 서버는 동시에 서비스하는 모든 스트림들에 대해 일정한 시간 주기를 단위로 필요한 양을 읽어내는 기법(CTL: Constant Time Length)이나 일정량의 데이터를 소비되어야 하는 시간까지 연속적으로 읽어내는 기법(CDL: Constant Data Length)을 적용하는 라운드별 액세스 정책을 사용한다[2].

동시수용 스트림수를 최대화하기 위한 방법은 디스크 시스템의 물리적인 성질, 디스크 배열의 구성방법 및 디스크 스케줄링 기법과 관련있다. VOD 시스템에서 스케줄링 기법은 동시수용 스트림들이 모두 동일한 데드라인을 가지도록 설계하는 공통 라운드 기반 스케줄링(Round-based scheduling)과 동일한 데드라인 없이 스트림 각각의 읽기 데드라인을 가지도록 설계하는 데드라인 기반 스케줄링(Deadline-based scheduling)으로 분류할 수 있다.

즉, 액세스 정책은 라운드별로 읽어야 할 데이터 크기를 정하며 스케줄링 기법은 스트림들의 서비스 순서를 결정한다. VOD 시스템에서는 이들 두가지 측면을 동시에 고려하여야 전체 버퍼크기, 재생시작 지연, 지연지터와 같은 요소들이 결정될 수 있다[4,

7,11,19,22-24,29,30].

[15]에서는 데이터 액세스 기법과 데이터 스케줄링 기법을 분리하여 고려하였다. 즉 데이터 액세스 기법으로는 CTL 및 CDL과 같은 기법들을 고려하였고 스케줄링 기법으로는 라운드 기반 스케줄링과 데드라인 기반 스케줄링으로 분리하였다. 이것은 타당한 결과로서 스케줄링 기법들이 VOD 시스템의 데이터 검색에 사용될 때는 액세스 기법이 CTL이냐 CDL이냐에 따라 스케줄링 기법 자체도 분류되어진다. 즉 SCAN, LOOK, GSS같은 기법들은 라운드 기반으로서 시간 개념이 가장 중요하다. 반면 EDF, SCAN-EDF와 같은 정책들은 블록의 데드라인 개념이 중심이 되므로 검색량이 일정한 경우라야 데드라인 개념이 의미가 있다. [15]에서는 CTL을 라운드 기반 스케줄링과, CDL을 데드라인 기반 스케줄링과 결합시키고 각각에 대하여 요구되는 최소 디스크 전달률과 버퍼 요구량을 계산하고 있다.

기존의 연구에서는 대부분 데이터 액세스 정책이나 디스크 스케줄링 정책에 대하여 따로 분리하여 적용하고 있다. 따라서, 멀티미디어 데이터 서비스를 지원하기 위해서는 데이터 액세스 정책과 디스크 스케줄링 정책이 통합된 알고리즘이 요구되고 있으나 이에 대한 연구는 미비한 실정이다.

본 논문에서는, 이러한 문제에 대하여 분석하고자 VOD 환경에서의 가변 비트율(VBR: Variable Bit Rate) MPEG 비디오 저장 및 검색 기법들의 성능에 대하여 기술하였다. [15]에서의 스케줄링 정책을 좀 더 일반화하여 각 액세스 정책들의 성능을 비교 분석하였다. 즉, 비디오 데이터에 대하여 디스크 배열 구조에 대해 각 정책을 적용하였을 때의 버퍼 요구량, 재생 시작 지연 및 최대 수용 스트림 수 측면을 고려하였다.

2. 관련 연구

2.1 디스크 배열 모델링

디스크 배열은, 물리적으로는 헤드가 원하는 트랙으로 움직여 원하는 데이터를 단일 채널을 통해서 전송해내는 기존의 구조를 따르지만(물리적인 관점), 논리적으로는 이들 단일 디스크들을 효율적으로 묶는 방법을 모색하여 디스크의 성능을 최대화하는 기술을 필요로 한다(논리적인 관점).

디스크 배열의 물리적인 관점에서의 성능 결정 요소들은 단일디스크의 액세스와 관련되는 탐색시간, 회전지연시간, 전송시간 및 부가적인 오버헤드 등이며, 논리적인 관점에서의 성능 결정 요소들은 디스크 배열을 구성하기 위해 필요한 여러 가지 정보인 디스크 배열의 멤버 디스크수, 동기화정도, 데이터 입상성 등이다.

2.1.1 물리적인 관점

물리적인 관점에서의 성능결정 요소들은 (1)과 같이 주로 단일 디스크에서의 데이터 액세스 과정과 관련된다.

$$T_{retrieval} = T_{seek} + T_{rotation} + T_{transfer} + T_{hd-switch} \quad (1)$$

여기서, T_{seek}는 탐색하고자 하는 실린더로 디스크 암을 움직이는 시간을 말하며, $T_{rotation}$ 은 탐색과정이 끝나고 원하는 데이터가 헤드 밑을 지나갈 때까지 기다리는 시간 그리고 $T_{transfer}$는 디스크 헤드가 디스크 플래터의 표면으로부터 데이터를 읽고 이 데이터를 시스템 버퍼나 디스크 제어 장치의 디스크 캐쉬 영역으로 전송하는 시간, 또 $T_{hd-switch}$는 헤드 전환시간을 의미한다.

탐색시간(seek time)은 디스크 성능에 영향을 미치는 가장 중요한 요소이다. 탐색 시간은 화일 검색의 접근 형태, 동시에 접근하는 다른 화일의 수 및 검색되는 디스크 블럭의 입상성(granularity)에 영향을 받는다. 탐색시간 모델은 (2)~(3)과 같이 트랙의 선형함수(linear function of tracks)로 나타내는 방법과 트랙의 비선형 함수(nonlinear function of tracks)로 나타내는 방법이 있다[14].

$$t(k) = a + bk, \qquad k \text{ 는 탐색거리} \quad (2)$$

$$t(k) = a + b\sqrt{k}, \qquad k \text{ 는 탐색거리} \quad (3)$$

디스크의 1회전 시간이 L일 때 회전지연 시간은 $[0, L]$ 사이의 균일 분포를 가지며 평균 $L/2$시간을 소요한다. 그러므로 회전지연 시간은 데이터 전송시간과 함께 디스크 회전속도와 관계가 있다.

한 디스크가 요구크기 s를 전송하기 위해서 걸리는 시간은 (4)와 같고, 이때 D_{disk}는 디스크 전달률이다.

$$T_{transfer} = \frac{s}{D_{disk}} \quad (4)$$

이러한 전송시간은 디스크 회전속도 및 디스크의 저장밀도 모델링 방법에 직접적인 영향을 받으며, 이

두 가지 요소는 디스크의 중요한 물리적 제한요소로 작용한다.

2.1.2 논리적인 관점

논리적인 관점은 주로 디스크 배열 구성과 관련된다. 그러한 요소들에는 디스크 배열을 구성하는 멤버 디스크 수, 디스크의 동기화 정도(ds : 한 디스크 그룹안에 들어가는 멤버디스크수), 스트라이핑된 데이터 블록 크기 등이 있다.

디스크 배열은 동기화 유무에 따라 동기화 디스크 배열과 비동기화된 디스크 배열, 이 둘의 장점을 결합한 혼합 구조로 분류할 수 있다. 이들 구조들을 성능 실험에서 고려하기 위해서는 혼합구조로 통일이 가능하다(그림 1).
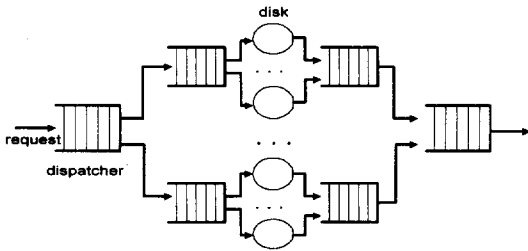
그림 1. 디스크 배열 모델

2.1.3 성능지수

VOD 서버에서는 각 스트림에 대해서는 요구하는 QoS 및 실시간성을 보장하고 시스템 전체적으로는 동시 스트림수를 최대화하는 것을 목적으로 한다. 본 절에서는 이중 최대수용 스트림수의 경향에 대해서만 기술하기로 한다.

먼저, 단일 디스크에서 동시에 서비스해야 하는 클라이언트수를 n이라 하면, 이러한 시스템은 (5)를 만족해야 한다.

$$\sum_{i=1}^{n}(T^{i}_{seek}+T^{i}_{rotation}+T^{i}_{transfer})\leq \min(T^{i}_{playback}) \quad (5)$$

여기서 저장된 데이터가 비디오라는 점을 고려하고 Zero-latency read 기법을 가정하여 회전지연시간을 0으로 두고 탐색시간을 최대값으로 두게 되면, Σ항을 간단하게 하여 최대수용 스트림수 n을 (6)과 같이 구할 수 있다.

$$n\leq \frac{T_{playback}-T_{\max seek}}{s/D_{disk}} \quad (6)$$

디스크 배열에서의 최대수용 스트림수는 배열 구성방법에 따라 조금씩 다르다. 우선 동기화된 디스크 배열의 경우에 전체 수용가능한 스트림수가 n'이고 동기화된 디스크수가 m이면 (7)을 만족해야 한다.

$$T_{\max seek}+n'\times\frac{s}{D_{disk}}\leq m\times T_{playback} \quad (7)$$

여기서 n'를 구하면 (8)과 같다.

$$n'\leq\frac{m\times T_{playback}-T_{\max seek}}{s/D_{disk}} \quad (8)$$

독립된 디스크 배열인 경우는 단일 디스크에서의 수용스트림수의 m배라고 생각하면 된다. 즉, 조건식은 (9)와 같고 전체스트림수는 n의 m배가 된다.

$$T_{\max seek}+n\times\frac{s}{D_{disk}}\leq T_{playback} \quad (9)$$

혼합구조인 경우는 그룹수(g)만큼은 비동기적인 디스크의 성질을 따르고 degree수 만큼은 배열의 동기적인 성질을 따른다. 따라서 각 그룹에서의 최대수용 스트림수를 n°라 할 때 주기는 degree수만큼 증가하므로 $\frac{m}{g}$배 늘어난다. 그러므로 전체적인 스트림수는 $g\times n^{\circ}$ 이다.

$$T_{\max seek}+n^{\circ}\times\frac{s}{D_{disk}}\leq\frac{m}{g}\times T_{playback} \quad (10)$$

단, $n'\geq g\times n^{\circ}\geq m\times n$. 따라서, 동기화된 디스크 배열의 경우가 최대수용 스트림수 측면에서 가장 우수하다.

3. 데이터 배치 방법(data allocation methods)

블록의 배치정책에는 파일 시스템이 보는 선형적인 블록 공간을 물리적인 디스크 시스템의 블록공간에 사상하는 정책과 스트림 내의 블록 배치를 파일 시스템의 선형적인 공간에 배치하는 정책이 있다. 전자에서는, 비디오 서버의 입출력 부하의 공간적인 특징은 임의적이며, 시간적인 특징은 주기적이라는 점이다. 후자의 경우에, 단일 디스크인 경우는 한 스트림 내의 블록 배치정책에는 임의적인 방법, 연속적인 방법과 두 가지 정책을 절충하는 연속된 블록의 거리를 제한하는 방법이 있다[5,9,13,15,18,29].

디스크 배열에서 동작하는 파일 시스템에 적합한 블록 배치정책은 단일 디스크의 정책과는 다르며, 이때 중요한 요소는 부하 균배 문제이다. 디스크 배열에 데이터를 조직하는 방법은 데이터 분산을 이용하

는 방법과 중복 메카니즘을 이용하는 방법이 있다. 데이터 분산은 각 멤버 디스크를 독립적으로 접근하는 방법과 스트라이핑을 사용하는 방법이 있으며, 전자는 각 멤버 디스크의 부하가 거의 같은 경우에 좋은 성능을 나타내며 후자는 부하의 균배, 전송의 병렬성, 접근의 병행성 측면에서 전자보다 우수하다.

중복 메카니즘은 디스크 배열내의 중복 정보의 종류, 범위, 위치를 지정하는 것으로, 반사를 이용한 기법과 패리티를 이용한 기법이 있다.

3.1 일반적인 데이터 배치기법

먼저 데이터 배치는 그림 2와 같이 액세스 빈도를 고려하지 않은 즉 부하 균배(load balancing)된다고 가정하고 구성할 수 있다.
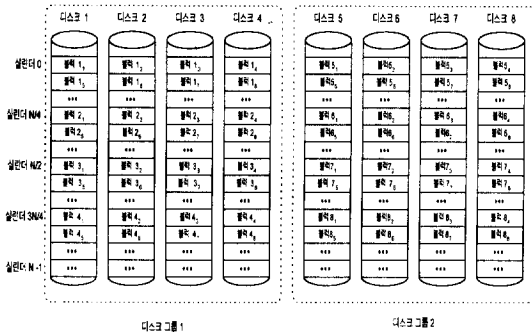
그림 2. 부하 균배되었을 때의 디스크 배열 구조

그림 2는 디스크 수가 16이고 비디오에 대한 재생 요청이 모든 디스크에 균형적으로 발생한다고 가정하였을 때 디스크 배열 내 블럭 배치 구조를 ds가 4인 경우를 고려하여 나타내었다(그림 2에서는 2개 디스크 그룹에 대하여만 나타냄). 가정에서 한 영화의 크기는 한 디스크 크기분으로 고정시켰기 때문에 동기화가 4인 경우는 4개의 영화가 4개의 디스크에 실린더별로 특정위치에 분산 저장된다는 것을 쉽게 알 수 있다.

3.2 비디오의 액세스 빈도에 따른 데이터 배치기법

인기있는 비디오(hot video)에 대해서는 액세스 빈도가 다른 것들에 비해 훨씬 높을 것이며 인기가 없는 비디오(cold video)는 거의 액세스를 하지 않을 것이다. 이것을 부하 균배시킨다면 수용 스트림수나 자원의 이용률이 낮아질 것이 예상된다. 그러므로 실제의 상황과 유사한 상황을 만들기 위하여 핫 비디오와 콜드 비디오를 구분하여 이러한 비디오에 대한 접근은 Zipf의 법칙(Zipf's law)를 따른다고 가정할 수도 있다[20]. 분포의 종류에는 단순 Zipf 분포(Simple Zipf)와 인기있는 비디오에 대해서는 조금 더 큰 요청 확률을 부과하기 위한 편중된 Zipf 분포(Skewed Zipf)가 있다(그림 3). 각 분포의 수식은 표 1과 같다. 여기서 n은 서버에 저장된 총 비디오의 수이며 i는 i 번째로 인기있는 비디오를 의미한다. 또한 θ는 왜곡정도를 의미한다.

표 1. Zipf 분포 구분

분포 / 항목	Simple Zipf 분포	Skewed Zipf 분포
액세스 빈도	$F_i = \frac{1}{i}(1 \le i \le n)$	$F_i = \frac{1}{i^{1+\theta}}(1 \le i \le n)$
액세스 확률	$P_i = \frac{F_i}{\sum_{j=1}^{n} F_j}$	$P_i = \frac{F_i}{\sum_{j=1}^{n} F_j}$

핫 비디오가 저장되어 있는 디스크에 대한 액세스 부하를 줄이기 위해 콜드 비디오와 같은 디스크에 스트라이핑시키는 방법을 사용할 수 있다. 그림 3에서도 관측할 수 있듯이 핫 비디오를 액세스할 확률은 콜드 비디오를 액세스할 확률에 비해 엄청나게 높다. 그러므로 스트라이핑시킬 때의 구조는 그림 4와 같은 스트라이핑 대안들을 고려할 수 있다.

이러한 대안들을 고려하였을 때 각 디스크 그룹에 걸리는 부하를 표 2에 나타내었다(단순 Zipf 법칙 적용). 수치적인 결과를 관찰해 보면 최적의 배치를 한
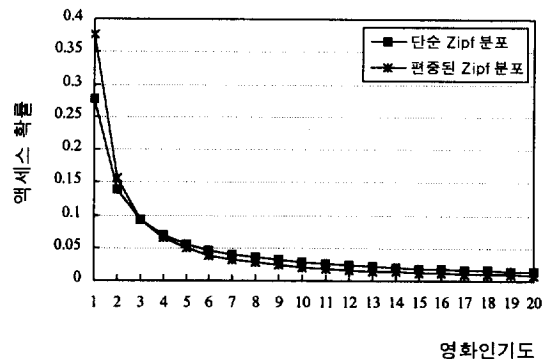
그림 3. Simple Zipf 분포와 Skewed Zipf 분포의 비교
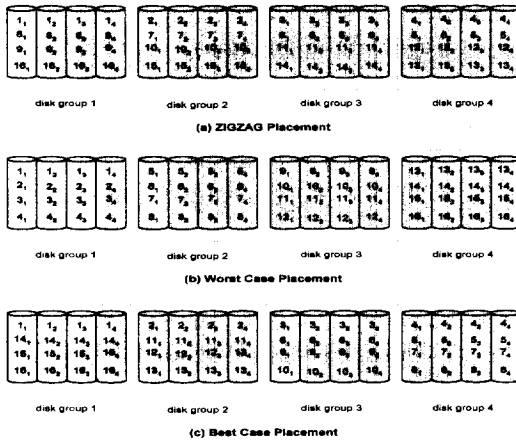
그림 4. 핫 비디오와 콜드 비디오를 고려한 데이터 블록배치 방법(ds=4)

표 2. 핫 비디오와 콜드 비디오를 고려한 배치방법에서의 디스크 그룹 선택빈도(디스크수=16, ds=4)

배치방법 / 디스크 그룹	ZIGZAG 배치	Worst Case 배치	Best Case 배치
1	6.14576	9.85984	5.68192
2	3.83136	3.00304	3.55504
3	3.13472	1.82384	3.36560
4	2.88816	1.31328	3.39744

다 하더라도 핫 비디오가 있는 디스크 그룹이 다른 그룹들보다 액세스 확률이 높음을 알 수 있다. 따라서 비디오의 사본을 복사하지 않고도 각 디스크에 부하균배를 이룰 수 있도록 여러 가지 효율적인 기법들을 사용할 것이 요구된다.

4. 데이터 검색(retrieval) 기법

4.1 액세스 정책

액세스 정책은 라운드별로 읽어야 할 데이터 크기를 정하는 기법으로, CTL 정책과 CDL 정책이 있다[2].

CTL 정책은 스트림 j에 대해 일정 재생구간을 가지고 가변길이 데이터 블록을 읽어내는 주기적인(periodic) 액세스 정책이다. τ_i^j인 i번째 라운드동안에 b_i^j인 데이터량이 저장소로부터 검색된다. CBR 데이터를 가정한다면 CTL 정책에서는 τ_i^j는 τ와 같고 $b_i^j = b^j$ 이다. 즉 CTL은 (11) 및 (12)와 같이 표현 가능하다.

$$W^j = \{b_0^j, b_1^j, b_2^j, \cdots\} \tag{11}$$

$$D^j = \{\tau, 2\tau, 3\tau, \cdots\} \tag{12}$$

그림 5는 CTL 기법 적용 시 시간별 누적 데이터량을 나타낸다. 라운드 기반 정책은 한 라운드동안에 다음 라운드동안 필요한 데이터량만 검색하고 남은 잔여시간은 휴지기간이 되므로 디스크는 라운드 중 최악의 경우를 지원하도록 설계되어야 한다.
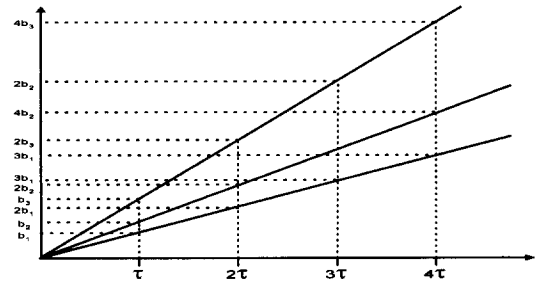
그림 5. CBR 트래픽일 경우 CTL 기법에서의 누적 데이터량

CDL 정책은 그림 6과 같이 디스크에 접근할 때마다 크기 b의 고정된 데이터를 읽어낸다. 그러므로 CDL 정책의 W와 D는 (13) 및 (14)와 같이 표현된다.

$$W^j = \{b^j, b^j, b^j, \cdots\} \tag{13}$$

$$D^j = \{d_0^j, d_1^j, d_2^j, \cdots\} \tag{14}$$
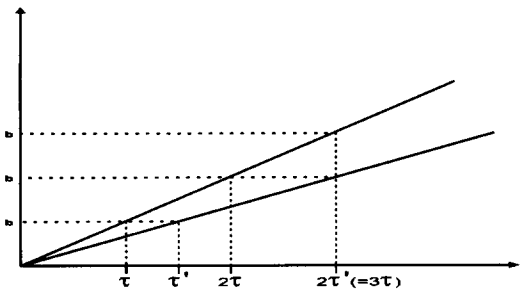
그림 6. CBR 트래픽일 경우 CDL 정책에서의 누적 데이터량

즉, 두 개의 데드라인 사이에 읽어야 할 데이터량은 동일하지만 데드라인 길이가 틀린 정책이다. 그러므로 스케줄링 정책은 데드라인을 고려한 EDF나 SCAN-EDF 정책 등이 함께 사용 가능하다.

4.2 스케줄링 기법

VOD 시스템에서 데이터 액세스는 일반적으로 라

운드 단위로 이루어진다. 그러므로 VOD 시스템에 스케줄링 정책을 적용할 때는 먼저 어떤 액세스 정책을 사용할 것인지 결정하여야 한다.

CTL은 한 서버에 들어오는 요청들에 대해 한 개의 공통된 주기마다 필요한 양을 읽어내도록 하는 반면, CDL은 가변적인 시간에 대해 동일한 크기의 데이터를 연속적으로 읽어내도록 고려하고 있다. 그러므로 CTL은 SCAN류의 스케줄링과 결합하고 CDL은 EDF류와 결합시키는 것이 가장 바람직하다고 판단된다.

그림 7에서는 라운드 기반 스케줄링의 예를 보여주고 있다. 세 개의 스트림(A, B, C)가 동시에 서비스 받고 있으며 그들의 공통된 주기는 t 단위로 진행된다. 그러므로 A, B, C는 라운드마다 모두 같은 데드라인을 가진다고 할 수 있다. 이러한 스케줄링에서는 같은 데드라인을 가진다는 점을 이용하여 탐색시간을 최적화 함으로써 성능을 향상시킬 수 있다.
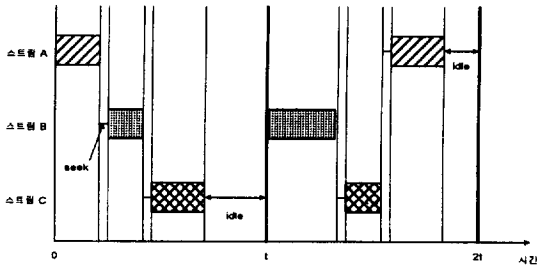
그림 7. 라운드 기반 스케줄링의 예

라운드 기반 스케줄링 기법으로는 LOOK, SCAN, RR 등이 사용될 수 있고, LOOK이 가장 최적의 성능을 나타내며 이것은 탐색 오버헤드가 가장 작다는 점에 기인한다. [19]에서는 이러한 라운드 기반 스케줄링의 일종으로 탐색시간을 최소화할 수 있는 SCAN 정책과 시작지연 및 총체적인 버퍼 소요량을 최소화할 수 있는 RR 정책의 장점을 결합하여 GSS (Grouped Sweeping Scheduling)를 제안하였다.

데드라인 기반 스케줄링(deadline-based scheduling)은 멀티미디어 데이터가 VBR적인 트래픽 특성을 가지므로 이로 인해 각 블록의 데드라인이 다른 점을 이용하기 위한 것이다. 그러므로 이 스케줄링을 이용하기 위해서는 데드라인이 모두 다르도록 조정할 수 있으며 VBR적인 트래픽 특성을 고려하여 각 주기마다 고정된 양의 데이터를 읽음으로써 데드라인을 다르게 조정할 수 있다. 그림 8에서는 스트림 A, B, C가 각각 고정된 양의 데이터를 다른 데드라인 안에 읽어내는 과정을 보여주고 있으며 이중버퍼를 사용한다고 가정한다.

여기서 C가 데드라인이 가장 작으므로 가장 먼저 블록1과 블록2를 읽은 후 A에 대한 블록을 읽는다. 그리고나서 데드라인을 비교해보고 데드라인이 가장 작은 B에 대한 블록을 읽었다. C에 대한 블록 4를 읽고는 휴지시간(idle time)이 생기는데 이것은 시스템이 각 스트림에 대해 이중버퍼를 사용한다고 가정했기 때문이다.
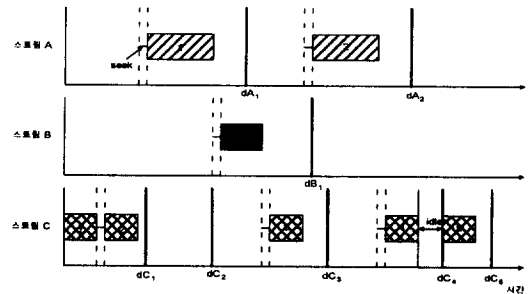
그림 8. 데드라인 기반 스케줄링의 예

5. 모의실험 및 성능 평가

5.1 모의실험 시스템 구성

전체적인 시뮬레이터 구조는 그림 9와 같다.
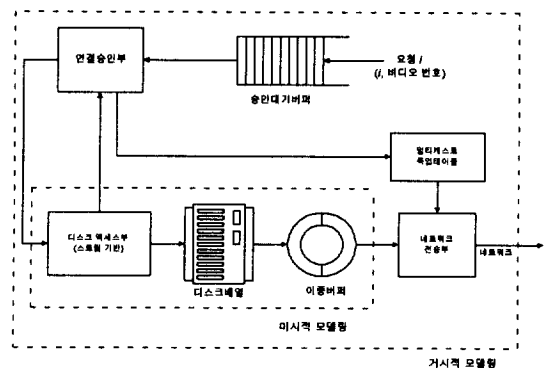
그림 9. 시뮬레이터 구조

그림 9에서 보는 바와 같이 미시적 모델링은 저장소의 액세스 성능과 관련한 사항만을 고려하였다. 즉 버퍼량과 네트워크 대역폭은 충분한 것으로 가정하

고 액세스와 관련된 성능분석을 수행하기 위해 요청 도착의 수는 모델을 사용하여 구성한 것이 아니라 모의실험 초기에 입력 파라미터로 결정하여 프레임 손실이 발생하지 않는 범위의 상한값들을 구하였다.

디스크에 대한 접근방법으로는 파일 시스템 기반(file system- based) 접근법과 스트림 기반(stream-based) 접근법이 있다[10]. 파일 시스템 기반 접근법은 요청-서비스를 반복하는 구조를 말하며, 스트림 기반 접근법은 한번의 요청 후에 연속적인 주기별 서비스가 이루어지는 구조를 말한다. 본 논문에서는 시스템이 스트림 기반 접근법을 사용한다고 고려하여 한번 승인된 요청에 대한 스트림은 서비스가 끝날 때까지 다시 요청이 없더라도 주기적인 서비스를 수행하는 것으로 가정하였다.

미시적 모델링에서는 디스크의 최대 성능을 평가하는 것이 그 목적이므로 스트림수를 모의실험 초기에 고정시키는 반면, 거시적 모델링에서는 가변 스트림수를 기본 원칙으로 하였다. 여기서 주의해야 할 것은 CTL(Constant Time Length)과 CDL(Constant Data Length)의 구분 범위는 거시적 모델링이나 미시적 모델링과 관련되는 것이 아니라 데이터 특성이 CBR인가 혹은 VBR인가와 관련되며 요구되는 재생률이 동종(homogeneous)인가 아니면 이종(heterogeneous)인가와 관련되는 것이다. 다음 표 3은 CTL과 CDL의 적용범위이다.

표 3. CDL 및 CTL의 적용범위

재생률 / 트래픽 특성	동종	이종
CBR	CTL	CTL, CDL
VBR	CTL, CDL	CTL, CDL

[2]에서 제안된 CTL, CDL의 의미는 한 요청 스트림에 대한 개념으로 표 3의 결과 중 CBR일 때 이종 환경에 적용된 CTL과 CDL은 원래 제안된 의미와 일치하는 것은 아니다. 그러나 CBR 트래픽 특성에서는 동시 스트림에 대한 서비스에서 CTL은 모든 스트림을 한 라운드로 묶는 것이 가장 바람직한 것으로 고려되며 CDL의 경우도 모든 스트림에 대해 한 라운드에서 읽는 크기가 같다고 가정하는 것이 바람직한 것으로 고려되므로 CBR 또는 VBR에 대한 언급없이 쓰기로 한다.

6.2 미시적 모델링에 대한 성능분석

6.2.1 모의실험 흐름도

미시적 모델링을 위한 모의실험의 흐름도는 그림 12와 같다.
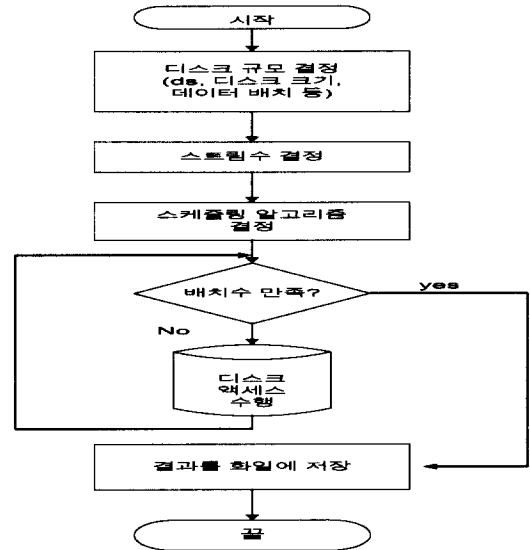
그림 12. 미시적 모델링을 위한 모의실험 흐름도

6.2.2 모의실험을 위한 가정

본 절에서는 디스크 배열 구조를 대상으로 한 미시적 모델링에 대한 모의실험 가정을 기술하고자 한다. 실험들은 스케줄링 정책 및 디스크 동기화 구조에 따른 최대수용 스트림수 및 디스크 전달단위의 경향 등에 초점을 맞추었다.

디스크 규모는 한 섹터는 512바이트, 한 트랙은 80개의 섹터로 구성되고 한 실린더는 20개의 트랙으로 이루어져 있으며 총 2752개의 실린더로 한 개의 단일 디스크를 구성한다. 그러므로 한 디스크의 용량은 2.098GB 정도의 용량을 가지는 것으로 하나의 비디오 내용을 수용하기에 충분한 용량이라 가정하고 기본적으로는 하나의 디스크가 하나의 비디오 용량을 가지는 것으로 고려한다. 또한 디스크 배열을 구성하는 멤버 디스크의 수는 16개를 가정하였으며 이를 바탕으로 *ds*를 고려하여 성능을 평가한다.

디스크 배열의 규모는 표 4와 같다. 여기서, 기본적으로는 하나의 디스크가 하나의 비디오 용량을 충분히 가지는 것으로 고려하여 *ds*에 따른 성능을 평가

표 4. 디스크 배열의 규모

섹터 (B)	트랙 (KB)	실린더 (KB)	디스크 (GB)	멤버 디스크수
512	40	800	2.098	16

한다.

본 논문에서의 실험을 위한 기본적인 가정은 다음과 같다.

① 고정률 검색인 경우만 고려(즉, 각 블록의 재생 주기는 일정하다.)
② 읽기 연산만 고려
③ 재생률 동일(1.5 혹은 3Mbps)
④ 연속 블럭 배치기법 사용
⑤ 한 영화의 크기는 한 디스크 크기로 고정
⑥ 블록의 크기는 트랙의 정수배수가 되도록 결정
⑦ 네트워크는 충분한 대역폭을 가지므로 전체 서버성능은 저장시스템 성능에 의해 좌우

블록 배치 방법은 기본적으로 연속배치 방법을 사용하였다. 단일디스크인 경우는 한 디스크 안에 한 비디오의 내용이 정확하게 들어가 있다고 가정하고 최대수용 스트림수 및 액세스 시간만을 고려하였다. 디스크 배열인 경우는 배열 구성방법에 따라 독립적인 디스크, 동기화된 디스크로 나누어 각각 단일디스크와 같은 블록배치 방법과 스트라이핑 기법을 사용하는 방법 등을 적용하였다.

6.2.3 디스크 배열의 성능분석

가. 탐색시간 모델의 영향

탐색모델의 사용이 최대수용 스트림수에 미치는 영향을 분석하기 위해 LOOK 스케줄링 알고리즘을 사용하여 단일디스크에 대하여 모의실험을 수행하였다. 탐색시간 모델별 최대수용 스트림수에는 변화가 없었고 읽어야 할 최소디스크 전달단위는 비선형 모델이 약간 우수한 것으로 나타났다. 비선형 모델이 우수하게 나타난 이유는 한 디스크에서 최대로 수용할 수 있는 스트림수가 5~6개이므로 평균 스트림간 트랙거리가 500 트랙 정도가 되어 선형모델보다 탐색시간이 작게 걸리기 때문이다(표 5 참조).

탐색시간모델을 디스크 배열에 적용시켰을 때 최대수용 스트림수 측면에서 비선형 탐색모델의 경우 선형탐색모델과 차이가 없음을 확인할 수 있었다. 이

표 5. 동일한 스트림수의 수용을 위한 선형 및 비선형 모델의 최소 디스크 전달단위(KB) 비교

선택모델 \ 스트림수	1	2	3	4	5
선 형	40	80	80	120	240
비 선 형	40	40	40	80	160

것은 최대수용 스트림의 경우 이동 트랙간의 평균지연이 모델에 따라서 별 차이가 없기 때문인 것으로 분석된다.

나. 스케줄링 알고리즘의 영향

CTL과 결합되는 스케줄링 정책으로는 SCAN류인 LOOK, RR, SCAN, GSS와 같은 알고리즘이 포함되며 이들의 성능은 버퍼요구량 및 재생 시작지연과 최대 수용 스트림수 측면에서 비교가 가능하다. 먼저 버퍼요구량 및 재생 시작지연은 표 6과 같이 분류가 능하다. 여기서, 라운드 길이 T_{round}와 한 스트림을 위해 필요한 단위버퍼크기 U는 (15) 및 (16)과 같다.

$$T_{round} \geq \sum_{j=1}^{m} (T^{j}_{seek} + T^{j}_{rotate} + T^{j}_{transfer}) \qquad (15)$$

$$U = \max_{1 \leq j \leq m} D^{j}_{disk} \qquad (16)$$

표 6. 알고리즘별 재생 시작지연 및 버퍼요구량

비교항목 \ 알고리즘	SCAN	GSS	RR
재생 시작지연	*Tround*	*Tround/g*	*Tround/m*
버퍼요구량	*2mU*	*(m+m/g)U*	*(m+1)U*

수용 스트림수 측면에서 최적의 스케줄링 알고리즘을 선택하기 위해 LOOK, SCAN, CSCAN, RR 알고리즘을 적용하였을 때의 *ds*별 최대수용 스트림수를 비교한 결과 그림 10과 같이 LOOK이 가장 우수하다. 그리고 동기화가 커질수록 최대수용 스트림수가 증가함을 알 수 있다.

그림에서의 결과를 비교해 보면 동일한 스케줄링 알고리즘을 적용한 경우, *ds*값이 증가함에 따라 최대 수용 스트림수가 증가하고, 스케줄링 알고리즘 측면에서 본다면 SCAN 및 LOOK이 가장 우수함을 확인할 수 있다. 스케줄링 알고리즘의 영향은 단일디스크에서보다는 디스크 배열에서 수용 스트림수가 증가하기 때문에 그 영향이 커짐을 확인할 수 있다. 그림 10에서 LOOK과 SCAN이 가장 우수한 성능을 보여주고 있다.
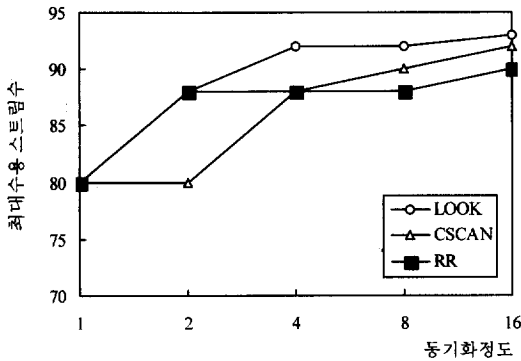
그림 10. SCAN류의 최대수용 스트림수 비교

이 알고리즘들을 비교하기 위해 최대 스트림수를 수용할 수 있는 최소 디스크 전달단위크기를 비교하였다. 그림 11에서는 LOOK이 SCAN보다 최소 디스크 전달단위 즉 최소 소요 버퍼 측면에서 약간 우수함을 확인가능하다.

위 결과들에 바탕을 둔다면 스케줄링 알고리즘은 가장 우수한 성능을 보여주는 LOOK으로 고정시킬 수 있다.
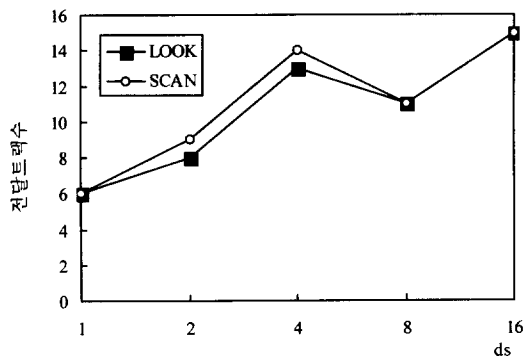
그림 11. *ds*에 따른 디스크별 최소전달단위(트랙갯수) 비교 (LOOK vs. SCAN)

EDF와 SCAN-EDF 등 스케줄링 정책에 데드라인 개념을 포함한 알고리즘들이 CDL 정책과 결합되어 사용된다. 이러한 정책의 버퍼요구량 및 재생 시작지연은 CBR 트래픽 환경 하에서는 SCAN류의 알고리즘과 동일하다. 즉 이중버퍼링인 경우 시작지연은 C이며 버퍼요구량은 $2mU$이다. 동종환경에서는 EDF는 RR과 같고 SCAN-EDF는 SCAN과 같으므로 그림 12와 같이 SCAN-EDF가 최대수용 스트림수 측면에서 EDF보다 우수하다.
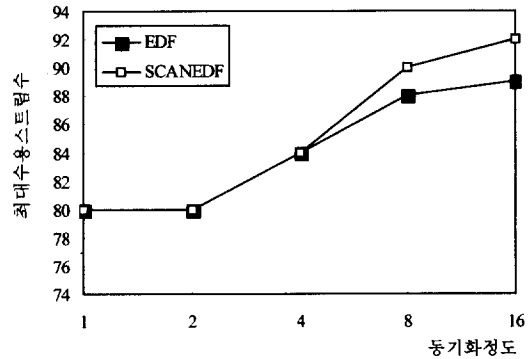
그림 12. EDF류의 최대수용 스트림수 비교

디스크 배열의 성능은 물리적인 제한 요소 외에 멤버 디스크수, 동기화 정도, 블록크기 및 스케줄링 정책에 영향을 받는다. 이들 요소들은 서로 상관성을 가지므로 본 논문에서는 멤버 디스크수를 16으로 가정하여 스케줄링 정책의 영향을 기준으로 *ds* 및 블록크기의 영향 등을 살펴보았다.

먼저 최적의 스케줄링 알고리즘을 선택하기 위해 LOOK, SCAN, EDF, SCAN-EDF, RR 알고리즘을 적용하였을 때의 *ds*별 최대수용 스트림수를 비교하여 그림 13에 나타내었다. 디스크 스케줄링 알고리즘에 대한 기술은 [16]을 참조할 수 있다.

그림 13에서의 결과를 비교해 보면 동일한 스케줄링 알고리즘을 적용한 경우, *ds*값이 증가함에 따라 최대수용 스트림수가 증가하고, 스케줄링 알고리즘 측면에서 본다면 SCAN 및 LOOK이 가장 우수함을 확인할 수 있다. 또한 RR은 동기화 정도가 낮을 경우에는 EDF나 SCAN-EDF보다 우수함을 확인할 수 있다. 스케줄링 알고리즘의 영향은 단일디스크에서
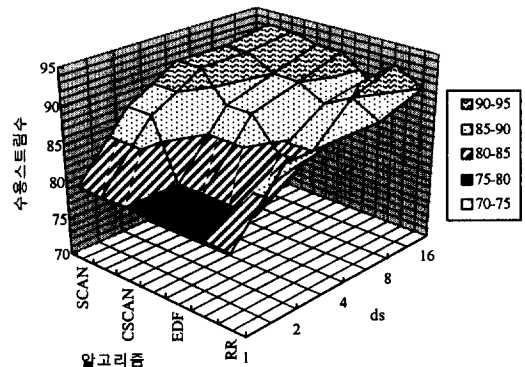
그림 13. 스케줄링 정책별 최대수용 스트림수 비교

보다는 디스크 배열에서 수용 스트림수가 증가하기 때문에 그 영향이 커짐을 확인할 수 있었다. LOOK과 SCAN이 가장 우수한 성능을 보여주고 있다.

위 결과들에 바탕을 둔다면 스케줄링 알고리즘은 가장 우수한 성능을 보여주는 LOOK으로 고정시킬 수 있다.

동기화 정도(ds)에 따른 최대수용 스트림수를 모의실험으로 얻은 결과는 그림 14와 같다. 그림에서도 확인할 수 있듯이 동기화가 커질수록 최대 수용스트림수도 늘어나는 경향을 가진다. ds가 16인 경우 최대 93 스트림까지 수용할 수 있음을 확인할 수 있다.

그림 14. LOOK 적용시 디스크 배열의 최대 수용 스트림수

5.3 거시적 모델링에 대한 성능분석

5.3.1 모의실험을 위한 가정과 알고리즘

거시적 모델링에서는 먼저 CTL/CDL 기법을 고려하였다. 실제로 CTL/CDL은 미시적 또는 거시적 모델링과 관련성이 있는 것이 아니라 디스크 스케줄링과 관련된다고 기술하였다. 그러나 이 기법이 전체 스트림 수용과 관련이 깊다는 측면에서는 오히려 거시적 모델링에서 다루는 것이 더 타당하리라 판단된다.

그림 15는 CTL 데이터 액세스 기법을 적용했을 경우의 모의실험 알고리즘을 나타낸다. 이 경우는 미시적 모델링에서 사용했던 방법(각 주기동안 한번씩 모든 스트림을 위한 블록을 검색해 주는 구조)과 같다. 다른 점은 재생률을 1.5, 3, 5, 8Mbps로 다양화하여 서로 혼합시키는 이종 환경으로 구현되었다는 점이다. CTL을 위한 스케줄링 기법으로는 LOOK을 사용하였다.

알고리즘에서 사용된 부호의 의미는 다음과 같다. S_{sector}: 섹터 크기, S_{track}: track 당 섹터의 수, S_{cyl}: 실린더에서의 트랙의 수, C_{disk}: 디스크에서의 실린더 수, n_{disk}: 디스크 수, ds: 디스크 배열의 동기화 정도, g: 그룹 수, no^{r}_{pi}: 각 재생률 별 스트림의 수, $S^{r}_{transfer}$: 스트림 전송률, avg_{access}: 평균 접근 시간, $max_{streams}$: 최대 허용 스트림 수, $roundlength$: 라운드 길이, no_{batch}: 배치 수

일반적인 승인제어 과정은 그림 16과 같다. 예를 들어 세 스트림을 수용할 여유가 서버에 있다고 가정하자. 요청도착 n-1과 n이 도착하면 먼저 승인제어부가 두 스트림을 위한 결정적인 성능보장 조건을 검사한다. 조건에 부합하므로 두 스트림은 바로 다음 라운드부터 서비스받을 수 있다. 그 다음 라운드에 도착한 n+1번째 스트림 또한 한 스트림을 더 수용할 수가 있으므로 수용 가능하므로 바로 다음 라운드부터 서비스 받을 수 있지만 n+2는 더 이상의 여유가 없으므로 거절된다. n+3 은 서비스중인 다른 스트림이 서버를 떠나기 전에 도착했으므로 거절된다. n+4는 자원이 해제된 후이므로 서비스 가능하다.

그림 17은 CDL 데이터 액세스 정책을 사용했을 경우의 모의실험 알고리즘이다. 그림 15와는 다른 구조로 여기서는 공통적인 주기의 개념이 없다. 그러므로 스케줄링도 데드라인으로 우선순위를 정하여 서비스를 수행하는 구조가 사용된다. do-loop가 한번씩 수행될 때마다 디스크 그룹별로 한 스트림만 서비스받을 수 있고 모의실험 시간도 각 스트림별로 개별적으로 관리된다. CDL에서는 prefetch가 가능하므로 버퍼 용량의 개념이 반드시 들어가야 한다. 즉 한 스트림을 위한 블록의 데드라인을 기준으로 했을 때 모의실험 시간 두 주기전보다 더 앞에 있으면 prefetching 할 수 없다.

그림 17의 알고리즘에서 사용한 부호 중 동일한 부호의 의미는 그림 15에서와 같고 다른 부호의 의미는 다음과 같다.

$S^{common}_{transfer}$: 공통 전송률, $deadline$: 블록의 데드라인, tf_{cycle}: 수행되지 않은 접근을 카운트하기 위한 변수

5.3.2 모의실험 결과

그림 18은 CTL 기법을 사용했을 때 ds가 2, 4, 16일 경우의 3Mbps 스트림수에 따른 1.5Mbps의 최

Input : Disk size(S_{sector}, S_{track}, S_{cyl}, C_{disk}, n_{disk}, ds, g)
$\sum(no^{r}_{pl}, S^{r}_{transfer})$ for every playback rate r(1.5, 3, 5, 8Mbps)
Output(file) : avg_{access}, $\max_{streams}$
Algorithm :
/* CTL access policy and LOOK scheduling are applied */
Decide *roundlength*;
Decide no^{r}_{pl} and $S^{r}_{transfer}$ for every playback rates;
for all concurrent streams
{
Allocate group, cylinder, track and sector position to j^{th} stream;
}
Gather streams for every disk group;
Sort streams of disk group i according to LOOK scheduling;
do {
roundlength++;
for every group i
{
for every streams j
{
Access for $stream_{ij}$;
}
current time = *roundlength*;
}
Check if no_{batch} is equal to MAX_BATCHES;
} while(no_{batch}<MAX_BATCHES for every group);
Save results in a file;

그림 15. CTL을 고려한 모의실험 알고리즘
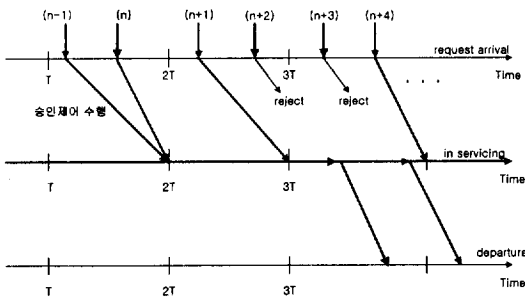
그림 16. 배칭기법이 적용되지 않을 때의 승인제어 과정

대 혼합비율을 그래프로 나타낸 것이다. 여기서, 스트림의 디스크 전달단위는 각 디스크별로 한 실린더 크기(800KB)가 되도록 할당하였다. 그래프에서 보면 동기화가 커질수록 ds가 증가할수록 최대수용 스트림수가 커지는 것을 알 수 있다. 그리고 ds =2인 경우와 ds =16인 경우를 비교해보면 ds =2 인 경우가 약간 굴곡이 지긴 하지만 큰 차이는 없는 것으로 판단된다.

그림 18에서는 schedulable region과 non-schedulable region을 각각 보여준다. schedulable region 안에서는 스트림들이 수용가능하다. 예를 들어 40개의 3Mbps 스트림과 80개의 1.5Mbps 스트림을 동시에 수용하는 것은 가능하지만 40개의 3Mbps 스트림에 대해 120개의 1.5Mbps 스트림을 수용하는 것은 non-schedulable region에 속하므로 수용이 불가능하다.

그림 19는 CTL과 CDL의 3Mbps와 1.5Mbps 스트림수의 혼합 비율을 비교한 그래프이다. 역시 디스크 전달단위는 디스크별로 한 실린더 크기가 되도록 하였다. CTL이 약간 우수(이유는 LOOK 스케줄링을 사용하므로 탐색 오버헤드에서 이득이 있음)하긴 하지만 거의 같은 곡선을 그리고 있다.

```
Input : Disk size( S_sector, S_track, S_cyl, C_disk, n_disk, ds, g)
        Σ(no^r_pi) for every playback rate r, S^comm_transfer
Output(file) : avg_access, max_streams
Algorithm :
    /* CDL access policy and EDF scheduling are applied */
    /* It is assumed that twin buffer were used for simulation */
    Decide no^r_pl for every playback rates and S^comm_transfer;
    for all concurrent streams
    {
        Allocate group, cylinder, track and sector position to j^th stream;
        Allocate roundlength_r to j^th stream;
    }
    Gather streams for every disk group;
    Sort streams of disk group i according to EDF scheduling;
    do {
        for every group i
        {
            for every streams j
            {
                if(current time>=( deadline_ij - 2× roundlength_ij))
                {
                    roundlength_ij++;
                    Access for stream_ij;
                }
                else
                {
                    tf_cycle++ to count access which is not performed
                            tf_cycle times;
                }
            }
            if( tf_cycle == no_i)
            {
                current time=( deadline_ij - 2× roundlength_ij) of stream j
                            with the earliest deadline ;
            }
        }
        Check if no_batch is equal to MAX_BATCHES;
    } while( no_batch<MAX_BATCHES for every group);
    Save results in a file;
```

그림 17. CDL을 고려한 모의실험 알고리즘
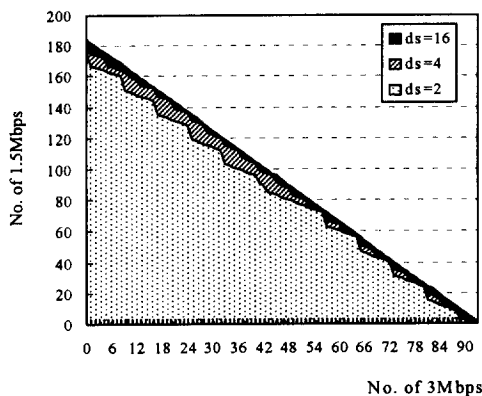
그림 18. CTL 기법에서 *ds*에 따른 3Mbps와 1.5Mbps 스트림수의 혼합비율
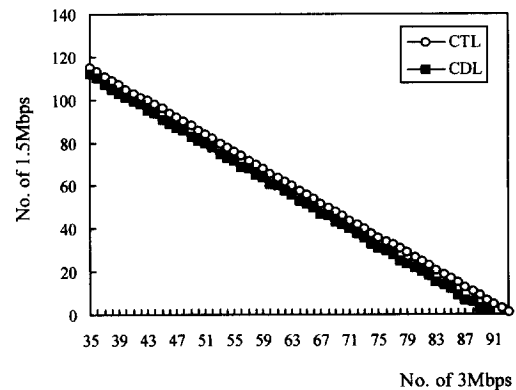
그림 19. CTL과 CDL의 3Mbps와 1.5Mbps 스트림수 혼합비율의 비교

그림 18과 19의 두 그래프를 다시 한번 고찰해 보면 CTL과 CDL 사이에 성능의 차이는 별로 나타나지 않는다. 하지만 CTL에서는 미리 자료를 읽어 서버 자원을 절약할 수 있는 경우는 한 라운드 길이로 제한되는 반면 CDL에서는 여러 라운드에 걸쳐 이루어질 수 있으므로 버퍼를 더 크게 늘릴 경우 수용 스트림수의 증가효과를 기대할 수 있을 것으로 판단된다.

6. 결 론

본 논문에서는 VOD 시스템의 저장소로 사용되는 디스크 배열의 성능을 비교분석하기 위하여 모의실험을 수행하였다. 스케줄링 정책을 좀 더 일반화하여 각 액세스 정책들의 성능을 비교 분석하였다. 즉, 비디오 데이터에 대하여 디스크 배열 구조에 대해 각 정책을 적용하였을 때의 버퍼 요구량, 재생 시작 지연 및 최대 수용 스트림 수 측면을 고려하였다. 성능분석을 통하여 단일디스크의 성능은 디스크의 물리적인 성능에 의해 주로 의존되는 것으로 판단된다. 탐색시간의 정도와는 관계없이 일정 스트림으로 제한되는 것을 확인할 수 있었으며, 스케줄링 정책별로도 단일디스크인 경우 수용 스트림수 측면에서는 큰 차이가 없음을 확인할 수 있었다.

이것을 디스크 배열로 확장했을 경우 스케줄링에 의한 성능의 영향이 커짐을 알 수 있었다. 디스크 배열은 단일 디스크의 물리적인 제한을 극복하고자 제안되어 여러 개의 단일디스크들이 하나의 디스크처럼 동작한다. 그러므로 비동기화된 디스크인 경우에는 단일디스크의 성능을 디스크수 만큼 확장한 것과 같으며, 혼합구조나 비동기화 구조보다는 동기화된 디스크 구조가 최대 수용 스트림수에서 더 우수함을 확인할 수 있었다. 회전지연 오버헤드도 디스크 배열에서는 중요한 요소로 동기화된 구조에서도 스트라이핑 단위가 트랙크기의 배수가 될 때 성능이 최적임을 확인할 수 있었다.

수용 스트림수에 대한 성능평가를 수행하였을 때 스케줄링 정책별로는 LOOK과 SCAN이 다른 알고리즘보다 효과가 있었으나 LOOK이 SCAN보다는 약간 더 우수하였다. 동기화 정도별로는 동기화가 커질수록 이득이 있었다. 그러나, 본 논문에서는 디스크 시스템 자체만을 고려하였기에 그러한 결과를 얻을 수 있었지만 이것을 네트워크와 연결시킬 경우는 우선 SCSI 용량도 고려해야 하고 네트워크 대역폭 및 버퍼 크기와 같은 자원량도 고려해야 하기 때문에 무한정 동기화 정도를 높일 수는 없을 것이다.

CTL 기법에서는 동기화가 커질수록 최대 수용 스트림수가 커지는 것을 확인할 수 있었고, 두 가지의 재생율에 대하여 스트림들을 수용할 수 있는 schedulable region을 구하였다. CTL과 CDL의 성능 비교에서는 CTL이 약간 우수하였으나 성능 차이가 크게 나지는 않음을 확인하였다. CTL에서는 미리 자료를 읽어 서버 자원을 절약할 수 있는 경우는 한 라운드 길이로 제한되는 반면, CDL에서는 여러 라운드에 걸쳐 이루어질 수 있으므로 버퍼를 더 크게 확장할 경우 수용 스트림수의 증가효과를 기대할 수 있을 것으로 판단된다.

앞으로 본 논문에서 기술한 CTL과 CDL 액세스 정책을 이용하여 네트워크와 연결된 VOD 서버 시스템을 고려하여, 보다 실제적인 MPEG 비디오의 저장 환경에서 비디오 검색 기법과 전송 기법을 연계한 방법에 대한 성능 평가 작업을 수행하는 것이 향후과제이다.

참 고 문 헌

[1] I. F. Akyildiz, S. Hrastar, H. Uzunalioglu, and W. Yen, "Comparison and Evaluation of Packing Schemes for MPEG-2 over ATM using AAL5," Proc. of ICC, pp. 1411-1415, 1996.

[2] Ed Chang and Avideh Zakhor, "Cost Analyses for VBR Video Servers", IEEE Multimedia, Winter '96, pp. 56-71, 1996.

[3] H. J. Chen, T. D. C. Little and Venkatesh, "A Storage and Retrieval Technique for Scalable Delivery of MPEG-Encoded Video," Journal of Parallel and Distributed Computing, Vol. 30, Issue 2, pp. 147-167, Nov. 1995.

[4] Mon-Song Chen, Dilip D. Kandlur and Philip S. Yu, "Optimization of the Grouped Scheduling (GSS) with Heterogeneous Multimedia Streams," ACM Multimedia 93, pp. 235-242, 1993.

[5] Ann L. Chervenak, David A. Patterson and

Randy H. Katz, "Choosing the Best Storage System for Video Service," ACM Multimedia 95, pp. 109-119, 1995.

[6] DAVIC Specification 1.0, DAVIC, Jan. 1996.

[7] Deepak R. Kenchammana-Hosekote and Jaideep Srivastava, "Scheduling Continuous Media in a Video-On-Demand Server," Proc. of Int. Conf. on Multimedia Computing & Systems, pp. 19-28, May 1994.

[8] Mengjou Lin, David Singer, Alagu Periyannan, "Supporting constant-bit-rate-encoded MPEG-2 transport over local ATM networks," Multimedia Systems, Vol. 4, pp. 87-98, April 1996.

[9] P. Lougher and D. Shepherd, "The Design of a Storage Server for Continuous Media," The Computer Journal, Vol. 36, No. 1, pp. 32-42, 1993.

[10] Guojun Lu, Communication and Computing for Distributed Multimedia Systems, Artech House, 1996.

[11] A. L. Narasimha Reddy and Jim Wyllie, "Disk scheduling in a multimedia I/O system", ACM Multimedia 93, pp. 225-233, 1993.

[12] Jian Ni, Tao Yang, and Danny H. K. Tsang, "CBR Transportation of VBR MPEG-2 Video Traffic for Video-On-Demand in ATM Networks," Proc. of ICC, pp. 1391-1395, 1996.

[13] P. Venkat Rangan and Harrick M. Vin, "Efficient Storage Techniques for Digital Continuous Multimedia," IEEE Trans. on Knowledge and Data Eng. Vol.5, No.4, pp. 564-573, Aug. 1993.

[14] Chris Ruemmler and John Wilkes, "Am Introduction to Disk Drive Modeling", IEEE Computer, pp. 17-28, March 1994.

[15] Sambit Sahu, Zhi-Li Zhang, Jim Kurose and Don Towsley, "On the Efficient Retrieval of VBR Video in a Multimedia Server" Proc. of MCS'97, pp.46-53, 1997.

[16] Ralf Steinmetz and Klara Nahrstedt, Multimedia : Computing, Communications, & Applications, Prentice-Hall, 1995.

[17] Fouad A. Tobagi et al. "Streaming RAID-A Disk Array Management System For Video Files," Proc. of 1st ACM Conf. Multimedia, pp. 393-400, 1993.

[18] Harrick M. Vin, Sriram S. Rao and Pawan Goyal, "Optimizing the Placement of Multimedia Objects on Disk Arrays," Proc. of Int. Conf. on Multimedia Computing & Systems, pp. 158-165, May 1995.

[19] Philip S. Yu, Mon-Song Chen, and Dilip D. Kaundlur, "Grouped sweeping scheduling for DASD-based multimedia storage management", Multimedia Systems 1, pp. 99-109, 1993.

[20] G. K. Zipf, Human Behavior and the principle of Least Effort, Addison-Wesley, 1949.

[21] 권택근, 최성연, "대화형 VOD 서비스를 위한 멀티미디어 서버의 설계", JCCI'96, pp.853-857, 1996.4.

[22] 박경호, 최양희, 김종상, "분산된 멀티미디어 중복 저장 서버의 정적 스케줄링", 한국정보과학회 봄 학술발표논문집, pp. 485-488, 1996.

[23] 박성수, 조동호, "멀티미디어 서버 시스템을 위한 스케줄링 알고리즘의 성능분석", 정보과학회 논문지, 제23권 제5호, pp. 467-477, 1996년 5월.

[24] 우회경, 박지철, 김종권, "VOD 서비스를 위한 멀티캐스트 스케줄링", 한국정보과학회 논문지 제22권 제3호, pp. 470-477, 1995년 3월.

[25] 이승섭, 이병욱, 장규환, "VOD 시스템을 위한 비디오 서버 접속", 한국통신학회지 제11권 제10호, pp. 920-933, 1994년 10월.

[26] 이주영, 신현식, 김동원, 김대웅, "VOD 서비스를 위한 비디오 서버와 대용량 저장 매체 설계", 하계통신학회 논문집, pp. 1641-1644, 1996.

[27] 장민, 유관희, 도상윤, 오옥균, 최진호, "VOD 시스템상 클라이언트-서버 구조에 대한 연구", 한국정보과학회 가을 학술발표논문집, 제22권 제2호, pp. 1037-1040, 1995.

[28] 조준성, 디스크 배열 기술을 이용한 실시간 디지털 오디오/비디오 저장 서버의 설계 및 분석, 공학석사 학위논문, 37면, 서울대학교 대학원 컴퓨터공학과, 1994년 2월.

[29] 황기태, 고성능 디스크 입출력을 위한 디스크 스

케쥴링 알고리즘, 공학박사 학위논문, 123면, 서울대학교 대학원 컴퓨터공학과, 1994년 2월.
[30] 허홍, 오해석, "ATM 기반의 분산 주문형 비디오 서비스를 위한 시스템 구성 및 프리젠테이션 모델", 한국정보과학회 가을 학술발표논문집 제22권 제2호, pp.1049-1052, 1995.

전 용 회

1978년 2월 고려대학교 전기공학과(공학사)
1985년 8월 ~ 1987년 8월 미국 플로리다공대 대학원 컴퓨터공학과
1989년 8월 미국 노스캐롤라이나주립대 Elec. and Comp. Eng. 석사(MS)
1992년 12월 미국 노스캐롤라이나주립대 Elec. and Comp. Eng. 박사(Ph.D)
1978년 1월 ~ 1978년 11월 삼성중공업(주) 근무
1978년 11월 ~ 1985년 7월 한국전력기술(주) 근무
1989년 1월 ~ 1989년 6월 미국 노스캐롤라이나주립대 Elec. and Comp. Eng. TA
1989년 7월 ~ 1992년 9월 미국 노스캐롤라이나주립대 부설 CCSP(Center for Comm. & Signal Processing) RA.
1992년 10월 ~ 1994년 2월 한국전자통신연구소 통신시스템연구단 선임연구원
1994년 3월 ~ 현재 대구가톨릭대학교 컴퓨터정보통신공학부 부교수
관심분야 : 차세대 인터넷, 멀티미디어 통신, 통신망 성능분석, 초고속망 프로토콜

박 정 숙

1994년 2월 효성여자대학교 전자계산학과(이학사)
1996년 8월 대구효성가톨릭대학교 전산통계학과(이학석사)
2001년 2월 대구가톨릭대학교 전산통계학과(이학박사)
2000년 ~ 현재 한국전자통신연구원 인터넷기술연구부 근무
관심분야 : 차세대 인터넷, 멀티미디어 통신, 통신망 성능분석, 트래픽 엔지니어링